

Ministero dell'Istruzione, dell'Università e della Ricerca  
Alta Formazione Artistica, Musicale e Coreutica

Conservatorio di Musica "G. B. Martini"  
Bologna

Tesi di  
Diploma accademico di I livello  
Corso di Musica Elettronica

Coreeno  
Un software per la generazione e la sintesi di strutture sonore

Tesi di  
Daniele  
Ponticelli  
Matricola 3711

Relatore: Prof. Francesco Canavese

Correlatore: Prof.ssa Annamaria Federici

Sessione Autunnale A.A. 2021/2022

## INDICE

Premessa.....	3
Capitolo 1 – Contestualizzazione.....	4
1.1 Breve storia degli algoritmi e dell’automatizzazione compositiva	
1.2 Microtonalità in Coreeno	
1.3 Approcci ricorrenti	
Capitolo 2 – Coreeno.....	17
2.1 Sequencer section	
2.2 Synthesis section	
2.3 Master section	
2.4 Possibili estensioni e miglioramenti	
Capitolo 3 – Sperimentazioni di utilizzo.....	41
Conclusione.....	43

## Premessa

Questa relazione viene proposta come elaborato scritto relativo a *Coreeno*, uno strumento realizzato con Max<sup>1</sup>. L'applicazione ha lo scopo di integrare metodi per la generazione di sequenze musicali con un sistema di sintesi sonora: dati certi parametri, il sequencer genera una serie di frequenze che vengono inserite in un processo per la sintesi additiva. Il progetto si compone di nove oscillatori a diverse forme d'onda e relativi parametri (modulabili), con la possibilità di lavorare in ambito tonale o microtonale su vari contesti ritmici.

Il principio alla base della realizzazione è la costruzione di uno strumento automatizzabile in alcuni dei suoi processi e con vari gradi di controllo e intervento, principalmente per esecuzioni in tempo reale. Un musicista meno esperto riesce a ottenere risultati complessi come la generazione di una melodia su una determinata scala, occupandosi di pochi parametri e con la possibilità di delegare certi procedimenti.

Questo strumento propone alcune soluzioni allo scopo di mantenere un equilibrio fra efficacia del controllo e semplicità di utilizzo, con un approccio volto alla gestione di molteplici eventi e di possibilità di intervento in tempo reale.

Il tool è nato nella concezione della musica temperata occidentale e si presta a un utilizzo a fianco di strumenti tradizionali del sistema temperato, ma offre varie finestre per un'interessante sperimentazione.

---

<sup>1</sup> Max è un ambiente di programmazione visuale di Cycling '74 di proprietà di Ableton

# Capitolo 1 – Contestualizzazione

In questo capitolo vengono raccontate alcune delle esperienze storiche in merito ai modelli di generazione musicale, nella scelta dei parametri quali frequenze, durate, ritmo, intensità e dell'integrazione di tutto questo con un approccio alla sintesi sonora.

## 1.1 Breve storia degli algoritmi e dell'automatizzazione compositiva

L'automatizzazione di alcuni dei processi produttivi o compositivi è oggi naturale e spesso presente nelle creazioni musicali: molti programmi permettono la generazione di sequenze di note su un'editor, si ha la possibilità di controllo dinamico dei parametri all'interno di una DAW<sup>2</sup> (automazioni), trasformazioni sull'audio e molto altro. Ci sono infinite possibilità nell'espressione musicale ed è sempre più importante riuscire a gestire una grande quantità di parametri e saper individuare quelli adeguati alla realizzazione delle proprie idee, applicando gli strumenti giusti nei contesti giusti. In questo senso è di aiuto conoscere le esperienze storiche all'interno dell'ambito ricercato.

Federico Simonetta, nel delineare *“brevemente le linee generali di una possibile storia del pensiero logico-razionale in musica”* nella sua tesi sulla modellizzazione musicale tramite catene di Markov (Simonetta, 2014), inizia esponendo gli studi di acustica delle antiche civiltà e le prime formalizzazioni a noi pervenute della scuola Pitagorica. In effetti la civiltà greca è fra le prime ad attribuire estrema importanza ai rapporti geometrici e ai fenomeni fisici in

---

<sup>2</sup> DAW è un acronimo per digital audio workstation ed è un sistema per lavorare con l'audio, spesso un software contenente funzionalità di ogni genere in ambito di registrazione, montaggio, riproduzione audio.

musica, avendo costruito un sistema musicale su determinate relazioni matematiche. Nel contesto del rapporto musica-matematica non si può parlare di vere e proprie composizioni automatizzate fino alla metà del Novecento, ma ci sono sicuramente delle esperienze particolari da citare, nel loro intento di raccogliere un insieme di regole per la composizione di musica in un modo più possibile indiretto.

Attorno al 1026 Guido D'Arezzo (991-1033) scrive il *Micrologus*, un importante trattato musicale nel quale è esposto un sistema musicale per associare testi poetici a note (Palisca, et al., 2001). D'epoca più recente è lo strumento *Arca Musarithmica* ideato da padre Athanasius Kircher (1602-1680), pensato per comporre musica senza alcuna conoscenza, esposto nell'ottavo libro della *Musurgia Universalis* (Chierotti).

Occorre considerare che la composizione di musica non è sempre stata intesa come qualcosa di immutabile, scolpita sullo spartito e di cui l'autore è la principale e più autorevole figura. Gli esecutori spesso avevano delle istruzioni e variavano entro un ampio grado di libertà e solo nel periodo classico si inizia a vedere il compositore come una figura molto importante e il frutto del suo ingegno come un valore immutabile. Béla Bartók (1881-1945) nei suoi *scritti sulla musica popolare* (Bartók, 1997) racconta delle esperienze di ricerca testimoniando una pratica, frequente nella musica popolare (anche europea), di improvvisazione e variazione sull'esecuzione di un brano. Tornando agli esempi storici, John Maurer (dal sito del CCRMA<sup>3</sup>) scrive che un ulteriore livello di astrazione è raggiunto con la nascita del canone in musica alla fine del quindicesimo secolo (Maurier, 1999), riferendosi all'imposizione di precise regole al compositore. È interessante che parli di "*livello di astrazione*", poiché di fatto automatizzare la musica significa perdere una parte di controllo del procedimento compositivo e porsi quindi a un livello più distaccato. Il canone è un metodo che, mediante un insieme di regole, permette

---

<sup>3</sup> Stanford "Center for Computer Research in Music and Acoustics"

di costruire delle voci a partire da una singola melodia; il compositore diventa quasi un progettista che si deve orientare fra alcune specifiche imposte.

Come già accennato, il vero processo di automatizzazione avviene solo nel ventesimo secolo, un periodo in cui si espandono molte delle crepe presenti in alcuni dei modelli adottati fino a quel momento e nei sistemi costituiti (non solo in musica). La dodecafonia e poi il serialismo rappresentano un'ulteriore fondamentale tappa nel pensiero musicale. Il serialismo integrale<sup>4</sup> (che ha radici nelle opere di Olivier Messiaen (1908-1992) ed ebbe il suo vero sviluppo con Pierre Boulez (1925-2016)) estende l'applicazione dell'organizzazione seriale ad altri parametri, non limitandosi alla frequenza. Rappresenta un punto di arrivo di un percorso inaugurato dall'approccio al cromatismo da parte di Richard Wagner (1813-1883) e proseguito tramite la Scuola di Vienna, fondata all'inizio del ventesimo secolo da Arnold Schönberg (1874-1951) e che ebbe tra i suoi più importanti allievi Alan Berg (1885-1935) e Anton Webern (1883-1945).

La storia delle composizioni automatizzate trova i primi sistematici approcci nell'ambito degli studi dell'università dell'Illinois, Urbana/Champaign, nelle persone di Martin Klein, Douglas Bolitho, Lejaren Hiller, Leonard Isaacson (Ames, 1987). Risulta abbastanza ovvio che certe applicazioni intensive dell'automatizzazione dei parametri in musica e una certa standardizzazione sarebbero potute avvenire solo con la comparsa dell'elaboratore. Esempi noti della scuola dell'Illinois sono: il celebre "Push Button Bertha (1956), brano derivato dall'uso di DATATRON, un computer sviluppato nell'università per comporre in automatico semplici composizioni musicali popolari, Illiac suite (1956), brano generato con l'ausilio di un

---

<sup>4</sup> La diffusione del serialismo ebbe una forte spinta con i corsi estivi di Darmstadt, a cui parteciparono varie personalità della musica contemporanea, come Luciano Berio (1925-2003), John Cage (1912-1992), György Ligeti (1923-2006), Luigi Nono (1924-1990), Henry Pousseur (1929-2009), Karlheinz Stockhausen (1928-2007)

altro computer (Illiac, per l'appunto), anch'esso progettato e costruito a Urbana, il MUSICOMP (per assistenza alla composizione) di Robert Baker (Ames, 1987).

James Tenney (1934-2006) fu uno studente di Hiller a Urbana e durante le sue esperienze studiò la composizione tramite computer e nel 1961 venne invitato ai Bell Laboratories<sup>5</sup>, dove Max Mathews (1926-2011) stava sviluppando un programma per la sintesi musicale. Tenney venne ispirato da alcune opere di Iannis Xenakis (1922-2001) come Pithoprakta e Achorriopsis descritte in articoli pubblicati dall'autore stesso. Queste ricerche produssero dei risultati interessanti poiché applicavano concetti specifici di estrazione delle frequenze, istruzioni e regole musicali per l'elaboratore e un uso di particolari strumenti matematici come le catene di Markov. Nelle ricerche all'università dell'Illinois gli esperimenti si ritenevano spesso conclusi con la realizzazione di un'opera e non si mirava a creare un vero e proprio modello compositivo. Già nei primi anni '60 in Europa invece si iniziarono a sviluppare programmi informatici con obiettivi a lungo termine e supportati da una struttura di idee consolidata. È il caso di Iannis Xenakis, importante compositore greco naturalizzato francese che fece ampio ricorso alla statistica e alla teoria della probabilità nei suoi lavori tanto da ideare un modello basato su processi stocastici. Egli faceva parte della cosiddetta scuola europea di cui si può citare Pierre Barbaud (1911-1990), fra i primissimi a presentare una composizione basata su computer, ma anche il compositore tedesco Gottfried Michael Koenig (1926-2021), il quale si avvicinò alla musica algoritmica europea dopo l'esperienza serialista di Colonia e che è stato associato per molti anni con l'istituto di sonologia di Utrecht (Ames, 1987). Merita menzione uno stretto collaboratore di Koenig, il compositore Otto Laske, che negli anni Ottanta ha co-fondato e co-diretto il NEWCOMP<sup>6</sup> con Curtis Roads (1951- ) e ha lavorato dai primi anni

---

<sup>5</sup> Centro di ricerca e sviluppo promotore di invenzioni rivoluzionarie apprezzabili anche in campo musicale

<sup>6</sup> New England Computer Arts Association

Settanta nell'ambito della psicologia cognitiva, della linguistica e delle intelligenze artificiali (Ames, 1987).

Dopo un periodo senza particolari innovazioni, negli Stati Uniti e in Canada emergono nuove figure che introducono l'utilizzo di programmi informatici pensati per esperienze in tempo reale. La diffusione di computer multiprogrammati offrì nuove possibilità di interazione fra uomo e macchina e si osservò la nascita di sistemi come GROOVE (sviluppato nel 1968 ai Bell Laboratories da Max Mathews e Richard Moore) che permetteva sintesi sonora e controlli in tempo reale (Di Nunzio, 2011), SCORE di Leland Smith del 1972, POD di Barry Truax del 1973, SAWDUST di Herbert Brun del 1976, SCED di William Buxton del 1978. Erano programmi di utilità accoppiati con possibilità di sintesi sonora, spesso costruita sui MUSIC4 e MUSIC5 di Max Mathews, ma più avanti anche su componenti hardware. Si notò negli anni una crescente interazione tra diverse discipline e ciò fornì nuove soluzioni compositive e fonti d'ispirazione (vene utilizzata ad esempio la teoria linguistica di Chomsky) (Ames, 1987). Oggi tanti concetti sono stati assimilati e vengono riproposti anche al di fuori dell'ambito accademico; grazie al progresso tecnologico è possibile ottenere a costi molto bassi quello che mezzo secolo fa era riservato a pochi facoltosi centri di ricerca. C'è una più semplice fruizione della musica e anche gli strumenti sono più accessibili. Si possono ottenere risultati importanti con pochi mezzi a disposizione, grazie alla capillare diffusione dei computer che hanno oggi una capacità di elaborazione che permette l'esecuzione in tempo reale e la sintesi sonora mediante software. Sono emerse nuove sfide nell'ambito del realismo e della credibilità della sintesi sonora e in tal senso le aspettative sono cambiate nel corso degli anni.

Un pianoforte offre la possibilità di suonare dei tasti ponderando l'intensità del suono e applicando delle modificazioni con i pedali e nonostante la sua semplicità (e in virtù della sua semplicità) è utilizzato da secoli e continuerà a essere utilizzato. Sotto un altro punto di vista,



questa semplicità non è affatto tale: ricostruire il timbro del pianoforte non è scontato e ancora di più non lo è simulare un'intera esecuzione. Il processo di apprendimento di questi fattori e la loro traduzione in un programma risulta molto complessa; perciò, la ricerca si è spostata nell'ambito delle intelligenze artificiali (Oracle, IA) (in particolare machine learning (Oracle, Machine Learning) e reti neurali (IBM, Reti Neurali)). L'intelligenza artificiale è un sistema che imita l'intelligenza umana per determinate attività e che si può migliorare con la raccolta di ulteriori informazioni. Questo tipo di approccio richiede l'utilizzo di una grande quantità di dati ai fini di istruire e dare un contesto alla macchina, attivando meccanismi articolati che vanno oltre il semplice ausilio alla composizione. È stato scritto un nuovo capitolo di Harry Potter basandosi sui libri precedenti (botnikstudios, 2018), ci sono programmi che generano storie (ilpost, 2022) o immagini (Google, Imagen) (OpenAI DALL-E-2), video (ictbusiness, 2022) a partire pochi elementi testuali o grafici. Tramite questo tipo di tecnologie è possibile programmare qualcosa che sia in grado di gestire una grande quantità di dati e imparare attraverso di essi.

## 1.2 Microtonalità in Coreano

Microtonalità è un concetto ampio che, in generale, indica un sistema di ripartizione dell'ottava per gradi diversi da quelli del temperamento equabile, cioè a intervalli minimi diversi dal semitono, talvolta equidistribuiti e altre volte di distanza variabile. "Microtonal music has no clear historic beginning or end. It has always existed, hiding behind different concepts or terms, or having no definition at all" (Ader, et al., 2020).

Il termine microtonalità può avere significati diversi: come uso di intervalli diversi dal semitono per suddivisioni più piccole all'interno della tonalità, come applicazione di sistemi di accordatura con una ripartizione dell'ottava diversa da quella standard oppure relativamente a sistemi appartenenti a musiche extra-europee. Osservando a un livello fisico, le frequenze appartengono a un insieme continuo, in contrapposizione ai raggruppamenti discreti (a gradini) che nella quasi totalità dei casi vengono utilizzati in musica. Sono stati proposti vari tipi di temperamenti e ognuno di essi offre dei miglioramenti ma ha anche degli svantaggi. Oltre alla classica accordatura occidentale si può citare il temperamento mesotonico, che si basa su un principio di ottenimento dei gradi della scala attraverso cicli di quinte con correzioni sugli intervalli di quinta, così che si trovino a un'intonazione naturale. Altri esempi sono la musica gamelan indonesiana, la musica classica indiana, la musica araba. Il temperamento equabile è d'uso principalmente occidentale, ma prima della sua recente adozione venivano usati sistemi differenti. L'uso dell'accordatura di un pianoforte, quindi, è tipica di certi popoli dell'occidente e nella storia di questi stessi popoli si trovano i primi cenni d'uso solamente a partire dal sedicesimo secolo.

Lo strumento discusso in questo elaborato intende affiancare strumenti tradizionali e offrire possibilità timbriche in un contesto di musica standard occidentale; nonostante ciò, sono state implementate delle sezioni per la sperimentazione in ambiti diversi, con parametri che permettono di discostarsi da un concetto di musica pop, soprattutto in relazione al ritmo e alla selezione delle frequenze. Come verrà esposto nel capitolo successivo, la generazione degli eventi in *Coreeno* può avvenire in situazioni alternative di BPM (battiti per minuto), con relative figure ritmiche tradizionali, o millisecondi, con finestre di tempo diverse. Il sequencer può inoltre lavorare in ambito tonale, cioè relativamente al temperamento equabile e alle scale tradizionali, oppure microtonale, con un pannello di generazione di gradi in scale

generate per ripartizioni diverse dell'ottava. Sarebbe un limite impedire a uno strumento pensato per la generazione di timbri e di strutture musicali di indagare ulteriori prospettive, in quanto con la nascita della musica elettronica e dei relativi strumenti di analisi del timbro e composizione del suono si sono sviluppati altri modi di osservare e intendere uno stesso oggetto. Il processo di costruzione del suono presente in questo strumento trae spunto anche dallo spettralismo, corrente che fonda le sue radici nel gruppo Itinéraire, nato in Francia nel 1973 e che coinvolgeva alcuni compositori come Murail (1947- ), Lévinas (1949- ), Tessier (1939- ), Dufourt (1943- ) e Grisey (1946-1998). Di fondamentale importanza è stato il supporto della tecnologia nell'analisi dell'oggetto sonoro e la correlazione tra immagine, suono e tempo. Secondo la definizione di Huges Dufourt la musica spettrale indica un lavoro di composizione che si esercita direttamente sulle dimensioni interne della sonorità, ossia sul timbro o spettro del suono (Manfrin, 2003). La musica elettroacustica e le tecnologie al suo seguito hanno offerto nuove possibilità in questo senso, anche solo nell'utilizzo dei software e apparecchiature presenti in tale contesto.

Una correlazione con il processo di creazione presente in *Coreeno* riguarda l'impiego di parametri appartenenti a un livello tipicamente macroscopico che possono essere usati per generare microstrutture o, più semplicemente, timbri. Il processo di assegnamento delle frequenze agli oscillatori o il controllo di parametri riguardanti l'intensità e la durata non sono infatti relativi a un singolo evento sonoro, ma è il legame fra queste proprietà e la loro evoluzione che contribuisce alla generazione di timbri particolari.

## 1.3 Approcci ricorrenti

Quando si osserva un pendolo oscillare si conosce il suo comportamento, non è di particolare interesse. Volendolo analizzare a livello fisico, però, si scoprirebbero molte formule e un mondo vastissimo. Così come nel pendolo ci si pone dalla parte dell'osservatore comune e non del fisico, ora si intende porre l'accento sull'ascolto. Nel sentire un suono di chitarra, non si pensa che esso sia la composizione di molte sinusoidi, che rispetto al sassofono queste sinusoidi siano presenti in minor quantità e con rapporti armonici consonanti; potremmo pensarlo, ma non è il ragionamento che un ascoltatore tende a fare. Si sa che un quartetto d'archi funziona bene assieme senza aver bisogno di analizzare lo spettro di ogni strumento perché attraverso esperienze pregresse si è capita questa cosa. Secondo John Maurer gli approcci all'uso del computer per la musica automatizzata possono essere generalizzati in tre aree: stocastici, basati sulle regole, intelligenze artificiali.

I processi stocastici (o aleatori) sono la descrizione dell'andamento nel tempo di una o più grandezze aleatorie, la cui evoluzione futura si sa descrivere in termini probabilistici (Treccani, Processo Stocastico). Questo tipo di approccio tiene quindi conto delle probabilità e delle distribuzioni statistiche, così da essere molto utile nel gestire grandi quantità di dati sapendo di poter ottenere un particolare effetto. La statistica, infatti, ragiona sui grandi numeri e per questo si presta molto all'evolversi del flusso musicale, che si è visto dipendere da infinite variabili. In questo ambito rientrano le catene di Markov e nonostante tale concetto non sia implementato nello strumento discusso in questa relazione, rappresenta un metodo estremamente utilizzato ed efficace nel contesto della generazione musicale. La proprietà che possiede un processo stocastico markoviano è quella di assenza di memoria: il valore assunto dalla variabile aleatoria allo stato futuro ( $n + 1$ ) dipende solamente dallo stato attuale ( $n$ ).

L'uso dei grafi e della rappresentazione a stati è particolarmente efficace poiché rende chiaro a livello visivo ciò che può essere complesso immaginare. Ogni stato ha una probabilità di transizione verso altri stati indicata da un numero fra zero e uno (corrispondente quindi a un valore percentuale). Ogni stato ha uno o più rami uscenti verso altri stati che indicano il possibile stato futuro (uno stato può avere anche sé stesso come stato futuro). Si usa una tabella, detta matrice delle transizioni, per rappresentare ogni probabilità di passare da uno stato attuale a un successivo.

Il secondo tipo di approccio fra quelli prima elencati è basato sui sistemi di regole, dai più semplici fino a vere e proprie grammatiche: la limitazione dell'insieme delle note estraibili alla scala diatonica nell'utilizzo del DATATRON (per le composizioni di Klein e Bolitho), le regole sul contrappunto nell'Illiatic Suite, la definizione di intervalli proibiti, quantità di consonanze o dissonanze, istruzioni sul ritmo, indicazioni sulle dinamiche d'esecuzione e così via. Questo tipo di approccio richiede di avere sistemi in cui un programma, con i propri algoritmi, è stato istruito a comportarsi in un determinato modo e potrebbe usare delle informazioni fornite tramite un database per eseguire un comportamento specifico.

Infine, le intelligenze artificiali: sono basate su un sistema di regole poiché di fatto costituiscono un programma informatico, con l'aggiunta riguardo alla capacità di apprendere e rinnovarsi secondo i dati forniti. John Maurer riporta l'esempio di David Cope e del suo sistema *Experiments in Musical Intelligence (EMI)*, il quale è basato su un'ampia banca dati contenente descrizioni di stili, regole, strategie compositive. Il programma, però, è anche in grado di generare una propria grammatica e un database di regole a partire da una serie di spartiti forniti in input. EMI è stato usato per comporre automaticamente musica che evoca stili di compositori d'altre epoche, come Bach, Mozart, Bartók, Brahms e molti altri.

## Capitolo 2 – Coreeno

*Coreeno* è uno strumento per la sintesi sonora e la generazione automatica di sequenze musicali, sviluppato in Max. In questo capitolo verranno descritte le funzionalità dell'applicazione con riferimento, qualora necessario, ad alcuni oggetti presenti nel software di Cycling '74 (Cycling '74, Documentazione Max). L'interfaccia è suddivisa in due sezioni principali, *sequencer section* e *synthesis section*, più una di controllo, *master section*. Il timbro viene costruito a partire dalla *synthesis section* dove sono presenti nove oscillatori organizzati in tre gruppi, uno per ogni tipologia di forma d'onda disponibile (sinusoidale, dente di sega, quadra), dei quali è possibile controllare alcuni parametri. Nella *sequencer section* si ha il controllo sulla generazione delle frequenze, durate, intensità, ritmo, registro.

L'aspetto peculiare del funzionamento dello strumento è la stratificazione delle sequenze melodiche, in un processo di trasferimento delle frequenze prodotte dal sequencer agli oscillatori tramite alcune tecniche di assegnamento allo scopo di ottenere sovrapposizioni verticali (armoniche) a partire da sequenze generate orizzontalmente (melodiche).

### 2.1 Sequencer section

Questa sezione è il *cervello* dello strumento, ci sono i controlli per la generazione delle sequenze ma anche la possibilità di impostare un input midi in tempo reale.

## Modulo ritmico

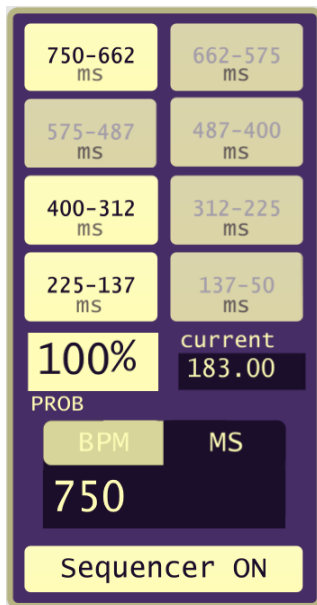


Figura 1  
Modalità millisecondi

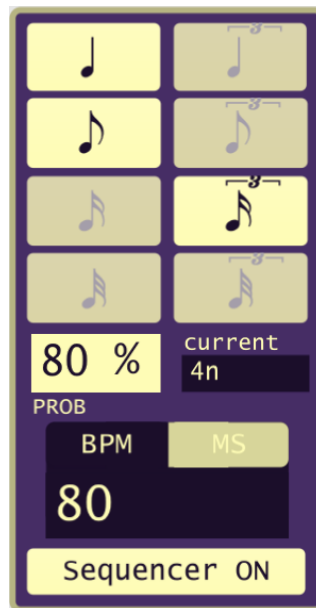


Figura 2  
Modalità BPM

La funzione di questo modulo è di generare degli inneschi da inviare al modulo per la selezione delle frequenze e da qui è possibile attivare o spegnere il sequencer. Quando è spento, ci si trova in modalità di midi input e ci sono due opzioni: in *single-mode* lo strumento è utilizzabile come un classico sintetizzatore da tastiera che risponde a messaggi standard di midi note (pitch, velocity). In *multi-mode*, la nota premuta sulla tastiera viene inserita nel processo di assegnamento agli oscillatori che sarà descritto nel prossimo sotto-paragrafo (modulo per la selezione delle frequenze). In questa seconda opzione di funzionamento vengono ignorati i valori di velocity e durata della pressione e la tastiera assume l'unica funzione di selezione delle frequenze.

Coreeno possiede un metronomo (o, in termini non musicali, un clock) interno di periodo  $T$  la cui pulsazione regola l'aggiornamento dei valori ritmici. Questo periodo può essere impostato alternativamente in millisecondi (Figura 1) o in BPM<sup>7</sup> (Figura 2). A ogni nuova pulsazione viene

---

<sup>7</sup> BPM e millisecondi di periodo  $T$  sono legati dalla formula  $T(ms) = BPM \div 60000$

scelto casualmente uno dei valori ritmici fra quelli selezionati e mantenuto per tutta la durata di  $T$ . A questo punto quel valore determina le altre sotto-pulsazioni all'interno del periodo di clock  $T$  e queste sono usate come innesco per la generazione delle frequenze per il modulo successivo. Se impostato in BPM, vengono visualizzate le figure ritmiche tradizionali in cui la semiminima ha esattamente periodo  $T$ . Se impostato in millisecondi, appaiono dei range di valori: quando uno di questi è estratto, viene scelto un valore casuale in quell'intervallo come periodo delle sotto-pulsazioni.

Nei casi particolari della semiminima in terzina e dei range di valori il periodo  $T$  non viene suddiviso in parti uguali come accadrebbe a un valore ritmico di periodo  $(T \div N)$ . Si consideri la situazione in cui per dei periodi di metronomo consecutivi viene estratto uno stesso valore ritmico: per periodi  $(T \div N)$  non viene avvertita nessuna irregolarità ritmica; negli altri casi, occorrono considerazioni particolari (Figura 3). Per il modo in cui è costruito *Coreeno*, a ogni nuova pulsazione del metronomo interno corrisponde la prima pulsazione del sottoperiodo, anche nel caso sia mantenuto il valore precedente. In quest'ambiente le suddivisioni ritmiche determinano la frequenza di generazione degli eventi e non la durata stessa dell'evento prodotto, che viene stabilita tramite l'involuppo (descritto nel dettaglio alla fine di questo paragrafo sulla *Sequencer section*).

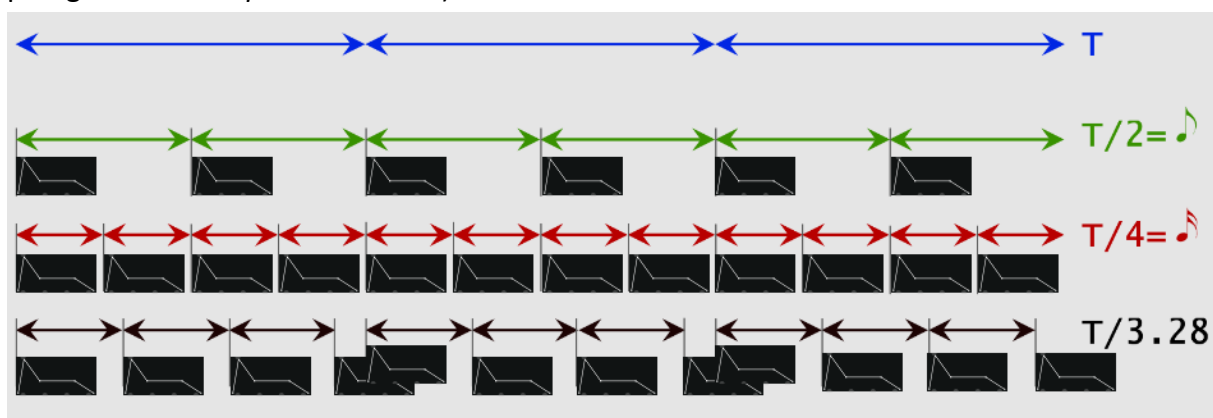


Figura 3

Ogni involuppo rappresenta la durata dell'evento. Sia nel caso della semiminima (in rosso) che nel caso di suddivisioni diseguali (in nero) si hanno quattro eventi all'interno di  $T$  ma il loro comportamento è diverso: nel secondo caso si ha una certa irregolarità ritmica e anche delle sovrapposizioni.



Queste sotto-pulsazioni vengono infine filtrate in base alla percentuale di riuscita indicata come PROB; a ognuna viene assegnato un numero da uno a cento e se il valore risulta maggiore della percentuale specificata, questo viene scartato.

## Modulo per la selezione delle frequenze



Figura 4  
Modalità tonale

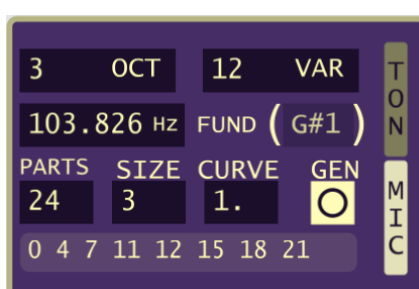


Figura 5  
Modalità microtonale

Il modulo ritmico produce degli eventi indirizzandoli al modulo per la selezione delle frequenze, il quale definisce l'insieme delle altezze disponibili, a partire dal minimo dato dalla fondamentale, lungo le ottave stabilite e per le sole frequenze appartenenti alla scala scelta. Ci sono due diverse modalità: tonale, basato sulla convenzione del sistema temperato in dodici semitoni, e microtonale, per un approccio a suddivisioni diverse dell'ottava.

Nella modalità tonale (Figura 4), dato un numero di ottave  $OCT = N_{OTT}$  e la lunghezza della scala  $L$ , il totale delle frequenze estraibili è  $TOT = N_{OTT} \times L$ . Se si indicizzano i  $TOT$  elementi di questo insieme con numeri da 0 a  $TOT - 1$ , allora 0 è la frequenza fondamentale, che in ottave successive ha indici  $(0 + L + OTT)$ , con l'ottava desiderata indicata come  $OTT \in [0, N_{OTT} - 1]$ .

Il parametro VAR (varianza) stabilisce la distanza massima possibile fra indici di due frequenze estratte successivamente, allo scopo di contenere i salti intervallari (se VAR=0 la frequenza resta costante).

Il *DETECTOR*, se attivo, permette di rilevare una scala dai tasti premuti su una tastiera midi in input: la nota più grave determina automaticamente la fondamentale, da cui poi è calcolata la scala. Se la sequenza è conosciuta (presente nel menu a tendina), viene visualizzato nell'interfaccia il nome della scala, altrimenti vengono mostrati i gradi che la compongono come numeri (da 0, fondamentale, a 11). Da notare che la rilevazione riguarda la scala, quindi se in input è premuta la stessa nota su più ottave ne verrà comunque considerata una.

Nella modalità microtonale (figura 5) il principio è analogo, con alcune peculiarità: la fondamentale è espressa in frequenza (si può comunque impostare una nota) ma la differenza principale riguarda la costruzione della scala, poiché in questo caso non si sceglie un modello preimpostato fra quelli disponibili, ma si genera ogni volta una sequenza. Rispetto alla modalità precedente vengono mantenuti i parametri di OCT e VAR, mentre i restanti sono sostituiti con *PARTS*, *SIZE*, *CURVE*, *GEN*.

Il tasto *GEN* (generate), basandosi sugli altri tre parametri, produce una scala visualizzabile in gradi numerici (principio analogo a prima con 0 come fondamentale) e come grafico modificabile nella parte dello *Scale diagram*, che verrà ripreso verso la fine di questo capitolo nella sezione dedicata.

*PARTS* è il numero di parti in cui viene suddivisa l'ottava e quindi stabilisce l'intervallo minimo, il microtono di riferimento, quello che nel sistema temperato è il semitono.

Per generare la scala, se  $PARTS = K$ , si può ottenere una successione combinando alcune fra le  $K$  frequenze disponibili (nel temperamento equabile sarebbero  $K = 12$  note disponibili). Si usano numeri da 0 a  $K - 1$  per indicare le possibili frequenze e le si suddividono in gruppetti

contigui di dimensione  $SIZE = S$  costante. Per ogni gruppetto viene poi scelto casualmente un elemento, in modo da ottenere una scala di lunghezza uguale alla parte intera di  $(K \div S)$ . Ad esempio, suddividendo l'ottava in PARTS=24 si ottiene un insieme di possibili frequenze indicizzabili da 0 a 23 all'interno di un'ottava. Impostando  $SIZE = 3$  si generano otto gruppi da tre elementi ciascuno, disposti per indici contigui, cioè  $[(0,1,2), (3, 4, 5) \dots (21, 22, 23)]$ ; la scala viene costruita scegliendo quindi per ogni gruppo una delle tre frequenze disponibili.

*CURVE* è un parametro utile a differenziare la composizione finale della scala: per valori maggiori di uno si ha una distribuzione dei gradi simil logaritmica (verso la fine dell'ottava) mentre per valori minori si ha una distribuzione simil esponenziale (verso l'inizio dell'ottava).

## Modalità di assegnamento delle frequenze

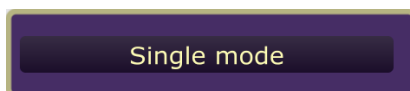


Figura 6  
In Single Mode spariscono i comandi presenti in Multi Mode

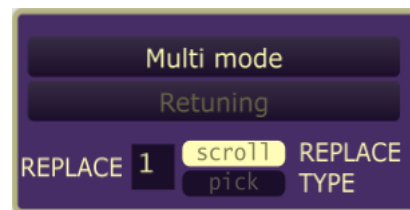


Figura 7

Le modalità di assegnamento agli oscillatori delle frequenze ottenute dal precedente modulo sono determinate dal pannello di Figura 6 o Figura 7 (modalità alternative).

In *single-mode*, una frequenza generata dal sequencer viene direttamente assegnata a tutti gli oscillatori, che avranno quindi stessa frequenza a meno di specifiche nei parametri della *Synthesis section*; in questo modo il sequencer produce una sequenza melodica.

In *multi-mode*, una frequenza generata dal sequencer viene inserita in un processo di assegnamento scelto fra le due tipologie: *scroll* e *pick*. In entrambi i casi, viene considerato l'insieme delle ultime nove frequenze generate in cui ognuna viene associata a uno dei nove oscillatori.

Si considerino le frequenze degli oscillatori come un vettore di nove elementi  $O = (O_1, O_2, O_3, O_4, O_5, O_6, O_7, O_8, O_9)$  che rispecchia la geografia dell'interfaccia (l'oscillatore più a sinistra è  $O_1$ , quello più a destra  $O_9$ ). Si considerino anche le ultime nove frequenze generate dal sequencer come un vettore  $F = (F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8, F_9)$  in cui  $F_1$  è la frequenza estratta più di recente e  $F_9$  è la più vecchia.

Nella modalità *scroll* l'associazione è biunivoca: la frequenza  $F_n$  è sempre assegnata a  $O_n$ , per cui una nuova frequenza estratta ( $F_1$ ) è assegnata all'oscillatore più di sinistra ( $O_1$ ), il cui vecchio valore è ora assegnato all'oscillatore alla sua destra, e così via. Da notare che a ogni ciclo il valore  $F_9$ , che viene sostituito con quello che prima era  $F_8$ , viene perso dalla memoria.

Nella modalità *pick*, l'associazione è simil casuale: la frequenza  $F_1$  viene assegnata a un  $O_n$  qualsiasi ma mai due volte allo stesso, se non dopo un aggiornamento completo del vettore  $O$ ; in altre parole, ogni oscillatore deve attendere sempre almeno nove eventi prima di essere nuovamente disponibile all'aggiornamento.

Nella modalità *multi-mode* le frequenze sono quindi distribuite in modo dinamico agli oscillatori e il risultato è che a ogni nuovo evento si ottiene un timbro vario e una fondamentale non sempre ben definita, in virtù delle rotazioni e delle differenze fra oscillatori (forma d'onda, intensità, detune, ecc.).

Il parametro *REPLACE* rappresenta il numero di sostituzioni che si vogliono effettuare tramite un singolo evento. Impostandolo a uno (default), per ogni nuovo evento si ha il comportamento fin qui descritto. Impostando un numero  $R$  diverso da uno, per ogni nuovo evento vengono generate  $R$  frequenze, velocizzando così il processo di sostituzione dei vettori di frequenze. Impostandolo a nove, di fatto, le due modalità *scroll* e *pick* hanno comportamento identico in quando per ogni evento viene aggiornato l'intero vettore  $O$ .

Se *RETUNING* è attivo, una modifica alla scala o alla frequenza fondamentale produce un aggiornamento complessivo del sistema, per cui viene simulata la generazione di nove eventi, sostituendo completamente i vettori delle frequenze; questo è pensato per avere subito disponibile il nuovo contesto senza dover attendere l'intero processo di sostituzione.

## Midi input



Figura 8

Qui vengono elencate le periferiche di input midi disponibili, che possono essere utilizzate per la rilevazione di scale o come input quando il sequencer è spento.

## Tabelle

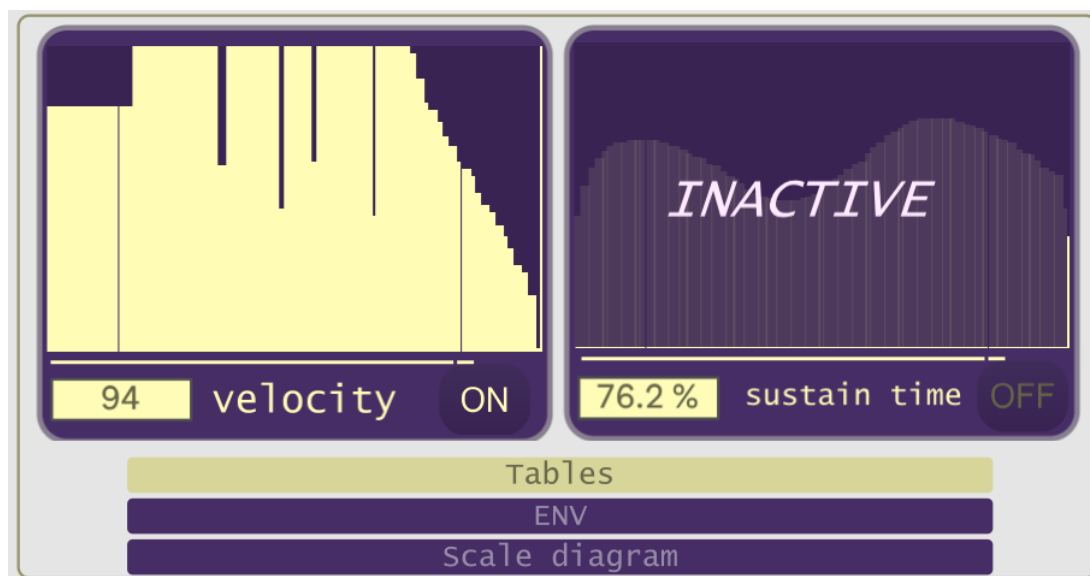


Figura 9

In questo caso è attiva solo una delle due tabelle. Il puntatore di lettura posto appena sotto allo spazio di disegno continua il proprio scorrimento anche quando una tabella è disattivata, ma mantiene l'ultimo valore memorizzato senza operare nuove letture

In questo blocco sono presenti delle tabelle contenenti settantacinque valori specificabili tramite puntatore disegnando all'interno dello spazio dedicato. Se attive, a ogni generazione di un nuovo evento del modulo ritmico viene letto il valore successivo in tabella, stabilendo così velocity e tempo di *sustain*<sup>8</sup> per quel particolare evento. Il *sustain time* viene inviato all'involuppo mentre la velocity, specificabile nell'intervallo (50, 127), determina l'intensità dell'evento associato. In generale, per stabilire la durata di un evento quando non si sta suonando in input midi, oltre al tempo di attacco, decadimento e rilascio dell'involuppo, viene considerato il *sustain time*. Questo tempo è espresso in percentuale rispetto al *domain time* e non è presente nel classico modello ADSR (dove il sustain è un parametro definito sulle ordinate per la funzione), ma in questo strumento di generazione automatica si è scelto di definirne un valore controllabile mediante la tabella.

## Inviluppo

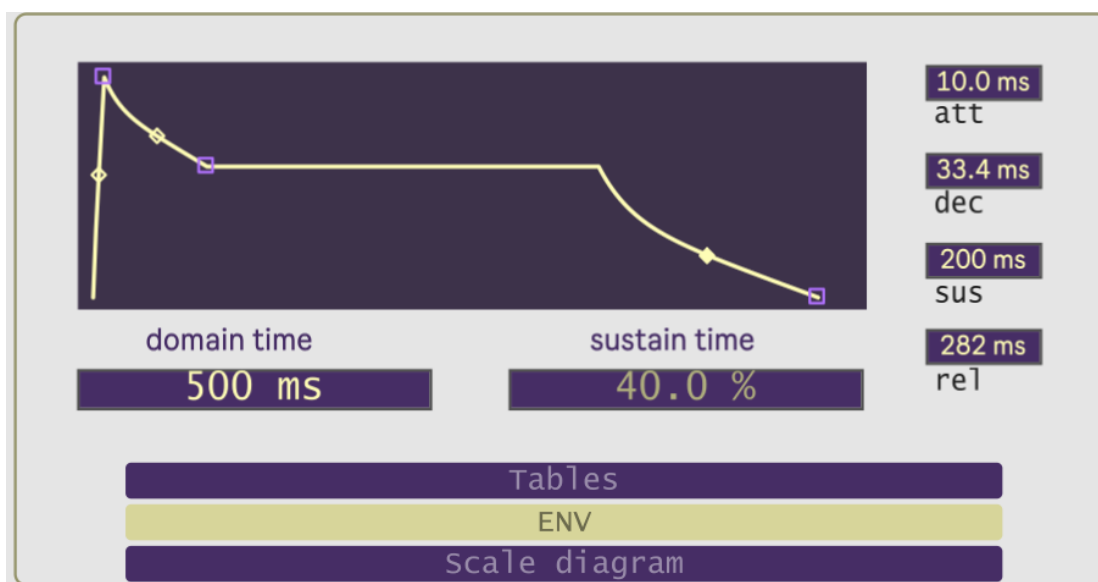


Figura 10

A fianco vengono visualizzati i valori effettivi per ogni segmento. La somma delle durate dei quattro segmenti determina la durata complessiva dell'evento.

<sup>8</sup> Si intende *sustain time* come tempo di mantenimento della pressione di un tasto sulla tastiera dopo l'evoluzione dei tempi di attacco e decay.

L'inviluppo segue il classico modello ADSR (attack, decay, sustain, release), con la peculiarità esposta in precedenza riguardante il tempo di sustain. Ogni segmento (attack, decay, sustain, release) ha una durata espressa in percentuale rispetto al *domain time*; tale rapporto è specificato tramite la tabella per quando riguarda sustain time e graficamente (nell'inviluppo) per gli altri valori<sup>9</sup>, trascinando il punto di inizio di ogni segmento tramite il relativo quadratino viola di Figura 10. Questo inviluppo viene assegnato all'oggetto sonoro corrente e quindi dal momento dell'inizio dell'evento sonoro in poi, quello manterrà nel tempo le informazioni specificate (eventuali modifiche dell'inviluppo durante l'esecuzione di quell'evento non incidono sullo stesso ma solo su eventi successivi).

## Diagramma della scala microtonale

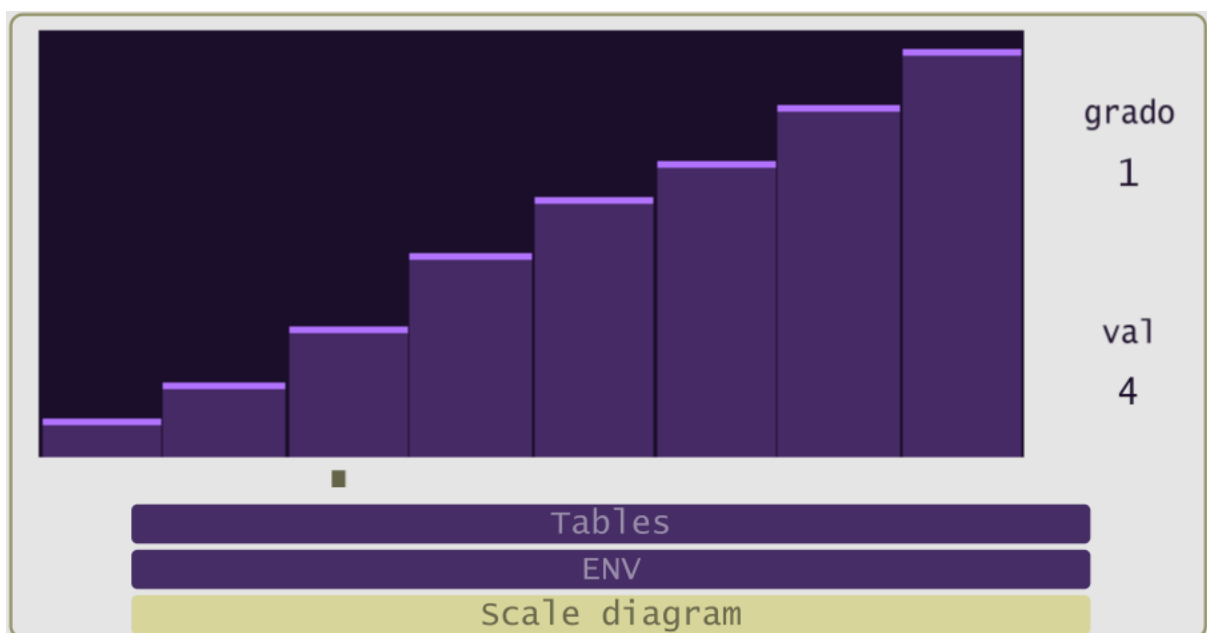


Figura 11

Quando viene modificato uno slider viene visualizzato, sulla destra, il grado della scala con il relativo valore. È possibile impostare valori in ordine non crescente ma verrebbe meno il senso del parametro VAR. Sotto agli slider è presente un quadratino che indica il grado attualmente estratto.

<sup>9</sup> È il comportamento dell'oggetto `live.adsr~` usato in questo progetto.

Questa schermata è un'interfaccia per la modalità microtonale, accessibile solo se si sta lavorando in essa e offre la possibilità di modificare la scala generata per adattarla alle proprie particolari esigenze. Queste modifiche vengono trasmesse anche alla lista di numeri rappresentante i gradi della scala presente nel pannello microtonale. In questo pannello, nella modalità tonale, vengono invece riportate le note della scala selezionata sulla tastiera.

## Considerazioni sulla sequencer section

L'associazione più semplice e diretta è in *single-mode*, ma è in *multi-mode* che *Coreeno* esprime al meglio le proprie potenzialità poiché l'idea iniziale alla base del progetto ruota proprio attorno a questa realizzazione.

Nella generazione degli eventi sonori le frequenze considerate per l'assegnazione agli oscillatori vengono valutate nel momento in cui è prodotto un nuovo evento. Per tutta la durata dell'involuppo, anch'essa valutata al momento dell'innesco, il vettore  $O$  relativo a quell'evento rimane costante.

In *Coreeno* si possono avere un massimo di venti istanze attive contemporaneamente; vale a dire che, per gli ultimi venti eventi, potrebbero esserci  $(O[1], O[2] \dots, O[20])$  vettori di frequenze associate ai nove oscillatori attivati in momenti diversi per un massimo di centottanta oscillatori insieme. Questo può avvenire, in particolare, se l'involuppo relativo a ogni evento è sufficientemente lungo da rimanere attivo per gli eventi successivi, cioè la durata dell'involuppo di  $O[E]$  (con  $E \in [0, 20]$ ) è maggiore della somma delle durate dei valori ritmici di tutti gli elementi successivi.



## 2.2 Synthesis section

Questa sezione ha tutti i controlli sulle sorgenti sonore dello strumento con diverse schermate per i compiti più specifici.

### Modulo gruppi e relative modulazioni

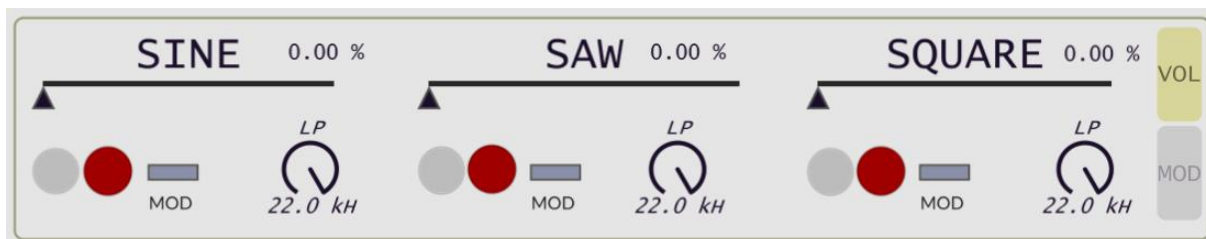


Figura 12 (Schermata VOL)

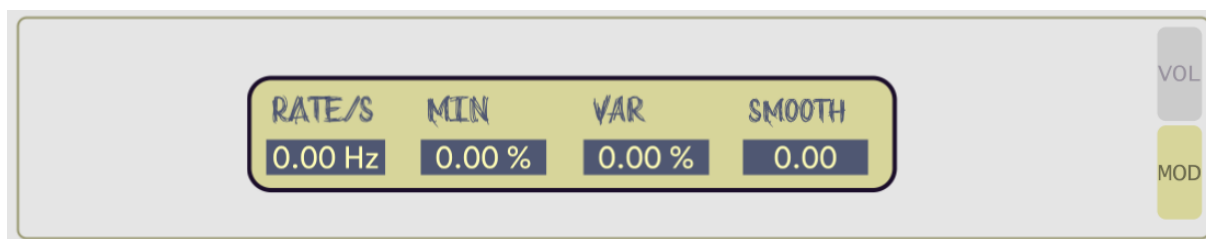


Figura 13 (Schermata MOD)

In *Coreeno* sono presenti nove oscillatori suddivisi in tre gruppi, uno per ogni forma d'onda (sinusoidi, denti di sega, quadre). Prima di essere dirette al master, le uscite degli oscillatori vengono instradate al proprio gruppo, che ne gestisce il volume ed è dotato di un filtro passa basso (oltre funzioni di mute/solo).

Questi volumi sono modulabili tramite un blocco apposito (Figura 13) che ne controlla l'andamento secondo i parametri specificati. Il *RATE/S* (rate al secondo) controlla la frequenza di aggiornamento del volume modulato, così che a ogni impulso si ottiene un nuovo valore a una distanza massima determinata da *VAR*. La transizione è più lenta e lineare per valori maggiori di *SMOOTH*: quando questo è uguale a zero, invece, il passaggio da un valore all'altro è immediato. *MIN* determina il valore sotto cui il volume non può scendere. Per attivare la modulazione è necessario accendere il bottone *MOD* (di Figura 12).

## Modulo oscillatori

In questo modulo ci sono i nove oscillatori fulcri della *Synthesis section* nonché le uniche sorgenti sonore dello strumento. Sono presenti tre schermate: *MAIN*, *MOD*, *FM* e cioè, rispettivamente, l'interfaccia di modifica diretta per gli oscillatori, le modulazioni sui parametri degli oscillatori e le impostazioni per la sintesi in modulazione di frequenza presente nei primi due gruppi (sinusoidali e dente di sega).

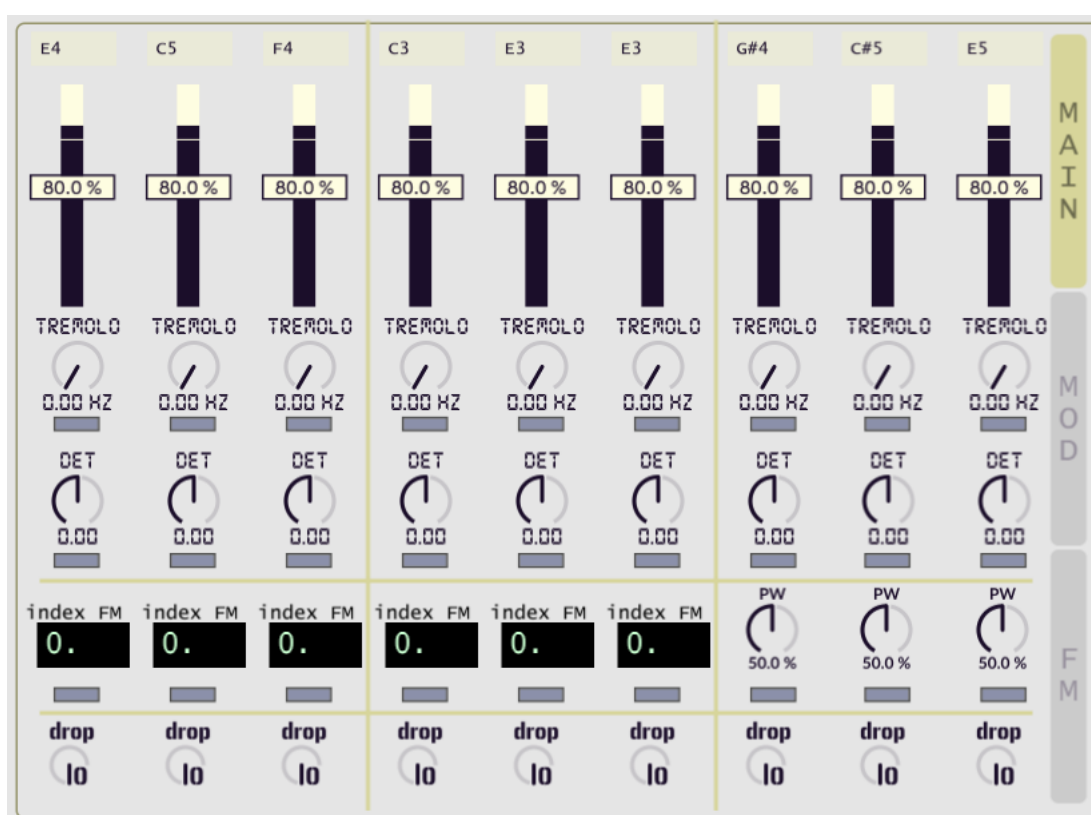


Figura 14

Da sinistra verso destra: 3 sinusoidi, 3 dente di sega, 3 onde quadre. Per ogni oscillatore sono controllabili gli stessi parametri (volume, tremolo, detune, drop) a eccezione di una riga contenente parametri caratteristici: indice di modulazione di frequenza per sinusoidi e dente di sega, pulsewidth per le quadre.

Nell'interfaccia principale *MAIN* (Figura 14) sono presenti i parametri di controllo degli oscillatori, che sono gli stessi per ognuno a eccezione di una riga di parametri caratteristici (*FM* e *PW*). Ci sono due tipologie di parametri che non sono modulabili, cioè *VOLUME* e *DROP*.

Il volume è gestito tramite uno slider che controlla l'intensità (in percentuale) dell'oscillatore. *DROP* indica il numero di ottave di trasposizione sul grave (la modifica avviene prima del detune). TREMOLO è costruito con un oscillatore sinusoidale a bassa frequenza che modula l'ampiezza generale dell'uscita. Nei parametri caratteristici, per l'onda quadra viene gestita la pulsewidth<sup>10</sup> mentre per le sinusoidi e i denti di sega è implementato un modulo di sintesi a modulazione di frequenza<sup>11</sup>.

---

<sup>10</sup> L'onda quadra assume valori alternati in  $-1$  e  $1$  di durata uguale. Questo in realtà avviene quando il suo duty cycle è al 50%; nella patch viene controllata la pulse width che rappresenta la durata dello stato in valore 1. Il duty cycle è invece il rapporto fra la pulse width e il periodo di pulsazione. L'onda, con duty cycle al 50% presenta solo armoniche dispari con ampiezza pari a  $1 \div N$ , dove  $N$  è il numero dell'indice della sinusoide nella serie armonica.

<sup>11</sup> La *FM* (frequency modulation) fu inventata da John Chowning alla Stanford University negli anni '60 e divenne popolare a partire dalla produzione del sintetizzatore DX7 della Yamaha (1983). Essa prevede un'onda (portante) la cui frequenza viene modulata da un'altra onda (modulante), controllata con un certo indice di modulazione. Questo tipo di sintesi produce risultati timbrici notevoli con poche risorse ed è ancora oggi largamente utilizzata, nonostante presenti inevitabilmente dei tratti evidenti di riconoscibilità (per questo spesso viene utilizzata assieme ad altre tecniche di sintesi, come nel caso di questo strumento).

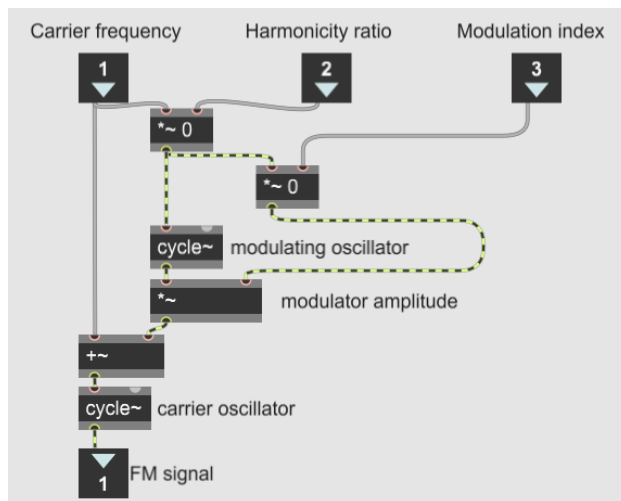


Figura 15  
SimpleFM, abstraction disponibile nella libreria standard di Max  
e basata sul modello di Jonh Chowning

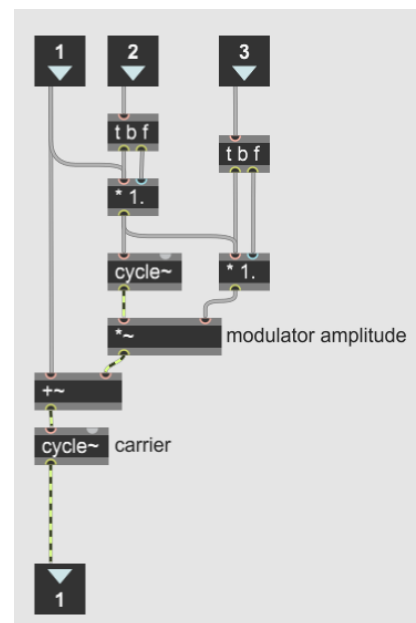


Figura 16  
Soluzione adottata per la FM del gruppo  
sinusoidi

La modulazione di frequenza per il gruppo di onde sinusoidali è stata implementata in *Coreeno* (Figura 16) ricalcando il modello di Chowning, a partire dall'astrazione *simpleFM* (Figura 15) presente nella libreria predefinita di Max. Nella soluzione adottata gli ingressi ricalcano il modello, per cui nel primo si trova la frequenza dell'onda portante, nel secondo il rapporto di armonicità (*harmonicity ratio*, che verrà discusso nel dettaglio come ultimo argomento del paragrafo) e nel terzo l'indice di modulazione.

Questo stesso modello di implementazione della modulazione di frequenza viene utilizzato per il gruppo delle onde a dente di sega, con delle particolari aggiunte. Rimane utile a scopo d'orientamento riferirsi all'astrazione *simpleFM* ma va considerato che in questo caso la portante non è una senoide, bensì un'onda a dente di sega; è quindi improprio riferirsi a questa realizzazione come FM semplice, sebbene sia quello il modello di riferimento.

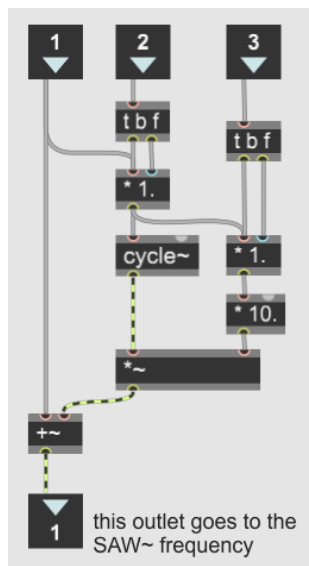


Figura 17  
Modulante per l'onda a dente di sega.  
Realizzazione ispirata dal modello della  
simpleFM

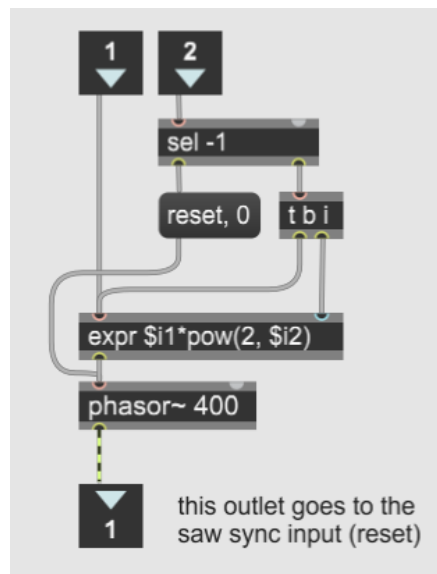


Figura 18  
Realizzazione dell'onda di  
controllo per sync input (reset  
dente di sega)

Rispetto alla realizzazione precedente l'indice di modulazione viene moltiplicato per dieci (Figura 17), di fatto estendendo il range dei valori possibili (e quindi sacrificando un po' di precisione nell'interfaccia controllo) poiché nella schermata *MAIN* questo parametro va da un minimo di zero a un massimo di dieci per entrambe le forme d'onda. Diverse sperimentazioni di utilizzo hanno portato alla definizione di questo fattore di riscaldamento dell'indice per obiettivi di efficacia dell'interfaccia e di risultato sonoro.

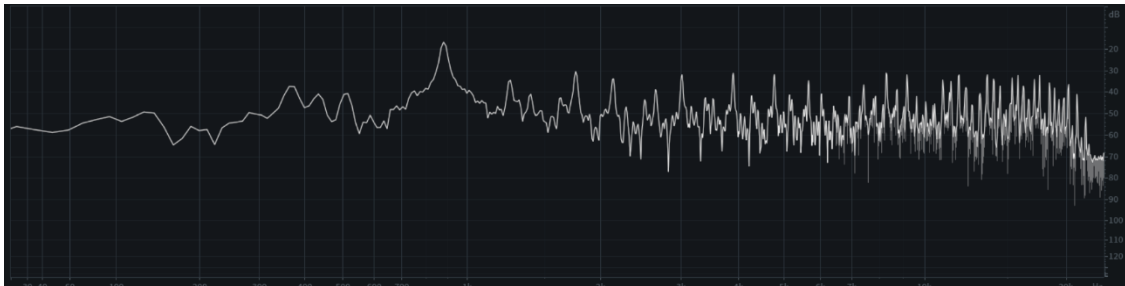
L'aspetto interessante dell'implementazione su quest'onda riguarda però la costruzione in Figura 18: in questa realizzazione, chiamando  $P$  la frequenza della portante e  $X$  il parametro in entrata al secondo ingresso, si ha un'onda a dente di sega di frequenza  $P \times 2^X$ , cioè a frequenza multipla della portante. Questa viene utilizzata per controllare un segnale di sincronizzazione: per costruzione<sup>12</sup>, quando essa attraversa da sotto il valore di 0.5 l'oscillatore principale viene risincronizzato, comportando una modifica del periodo e quindi

<sup>12</sup> È il comportamento dell'inlet di destra dell'oggetto saw~ di Max

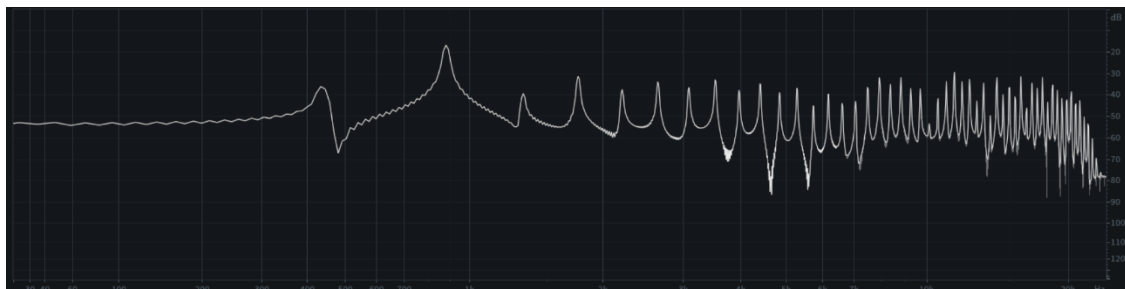
della frequenza. Per il modo in cui è costruito mediante rapporti di ottave, si ha quindi un effetto di distorsione spettrale, riportato di seguito e descritto a titolo esemplificativo, dal quale risulta con maggiore evidenza un'attenuazione delle frequenze più basse e un'enfatizzazione non lineare delle frequenze più acute.



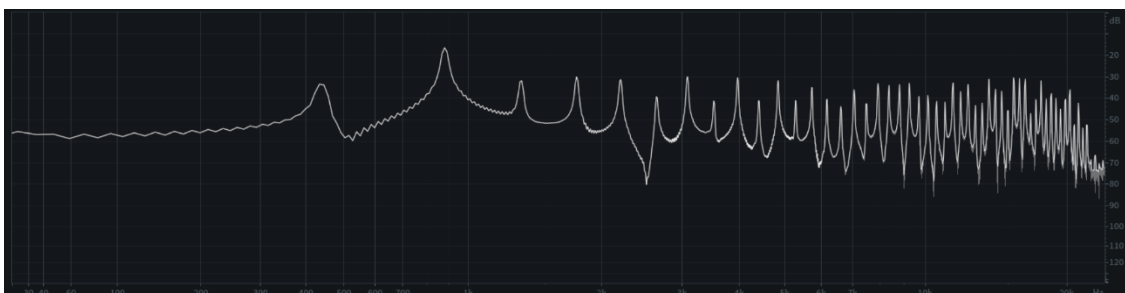
*Figura 19*  
*Spettro di un'onda a dente di sega a 440Hz*



*Figura 20*  
*Spettro di un'onda a dente di sega a 440Hz modulata in frequenza da un'onda sinusoidale a 440Hz con indice di modulazione a 50*



*Figura 21*  
*Spettro di un'onda a dente di sega a 440Hz modulata in frequenza da un'onda sinusoidale a 440Hz con indice di modulazione a 50 e onda di controllo per il reset a 440Hz*



*Figura 22*  
*Spettro di un'onda a dente di sega a 440Hz modulata in frequenza da un'onda sinusoidale a 440Hz con indice di modulazione a 50 e onda di controllo per il reset a 880Hz*

Per come è stato costruito lo strumento ci si trova qui in una situazione particolare per cui il reset del periodo dell'onda modulata viene controllato da un'onda che ha frequenza multipla della fondamentale e si ottiene un effetto di rimozione di alcuni contenuti in bassa frequenza. In Figura 20 si osserva la presenza delle componenti generate tramite modulazione di frequenza, ottenendo uno spettro più complesso di quello dell'onda originale di Figura 19. Come si nota invece in Figura 21, con l'aggiunta dell'onda di controllo per il reset del periodo si ha una riduzione di alcune componenti in bassa frequenza. Questo effetto è più evidente per frequenze maggiori dell'onda di controllo come si può osservare in Figura 22, dove la frequenza è raddoppiata rispetto alla figura precedente.

La seconda schermata presente nel blocco centrale di questa sezione è *MOD* (Figura 23), utilizzabile per la modulazione dei parametri su *MAIN* che hanno l'interruttore blu chiaro posto sotto di loro. Tale interruttore serve per attivare/disattivare la ricezione di questa modulazione.

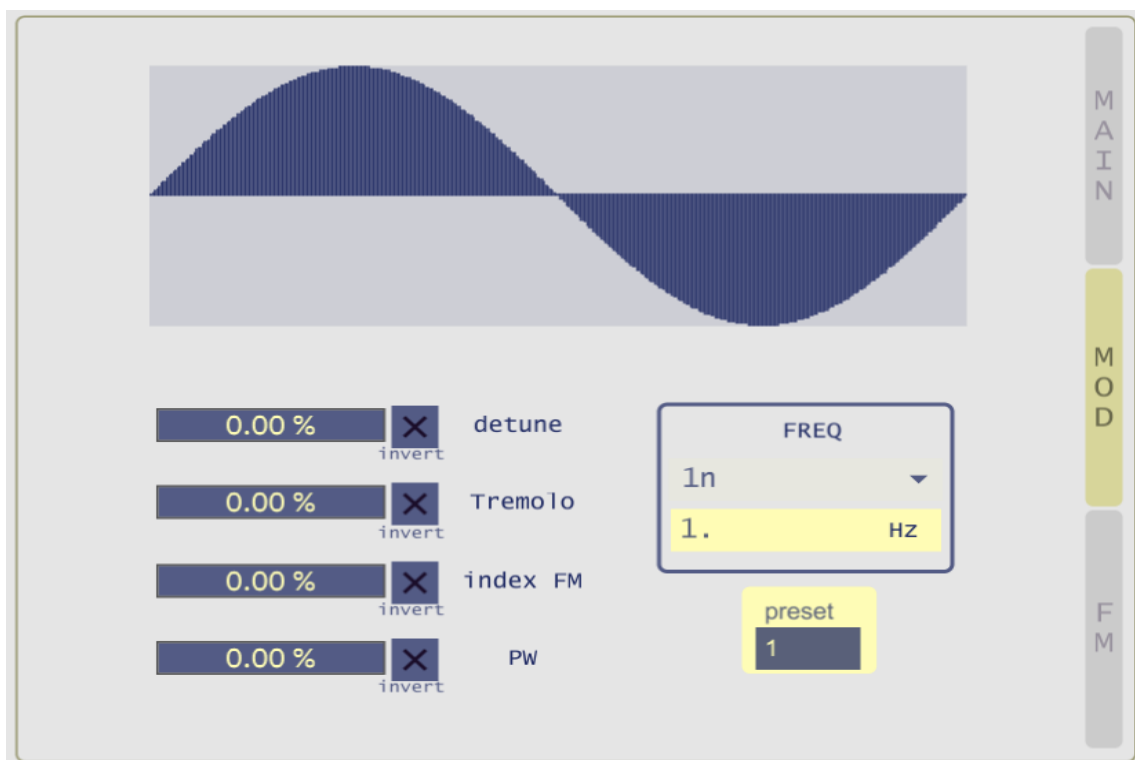


Figura 23

Nel pannello in figura 23 è presente uno spazio per disegnare l'onda modulante ed è possibile indicarne la frequenza in Hz o in relazione al metronomo interno ( $4n$  coincide con il periodo  $T$  descritto all'inizio del capitolo); quindi, per ogni gruppo di parametri è possibile specificare la quantità di modulazione desiderata e, opzionalmente, invertire l'onda prima di inviarla ai parametri da modulare. Questa realizzazione prevede cioè che venga disegnata una sola onda e che questa sia utilizzata come traiettoria della modulazione.

La modulazione di un parametro è attiva quando viene acceso il bottone posto sotto di esso che in questo modo segue l'andamento dell'onda nel tempo, per i valori indicati in percentuale relativamente al suo gruppo. La funzione disegnata ha, di base, valori disegnabili fra  $-1$  e  $1$ ; specificando la quantità di modulazione di ogni parametro si va a moltiplicare quest'ultimo valore alla funzione, in un'operazione di riscaldamento dell'onda sull'asse delle ordinate. Viene quindi disegnata una sola onda che si comporta diversamente per ogni gruppo di parametri in riferimento a diversi massimi e minimi sulle ordinate (la frequenza dell'onda è uguale in ogni caso). La quantità di modulazione è indicata in percentuale per motivi di trasparenza e rappresenta valori adeguati al parametro modulato. Inoltre, se tramite il processo di modulazione un parametro assume valori sotto il suo minimo o sopra il suo massimo, questi vengono ignorati e si mantiene per tutto il periodo di tempo il relativo valore minimo o massimo. Questo approccio, frutto di risultati sperimentali, offre un compromesso fra efficacia di intervento e semplicità di utilizzo, in linea con i principi alla base della costruzione dello strumento.



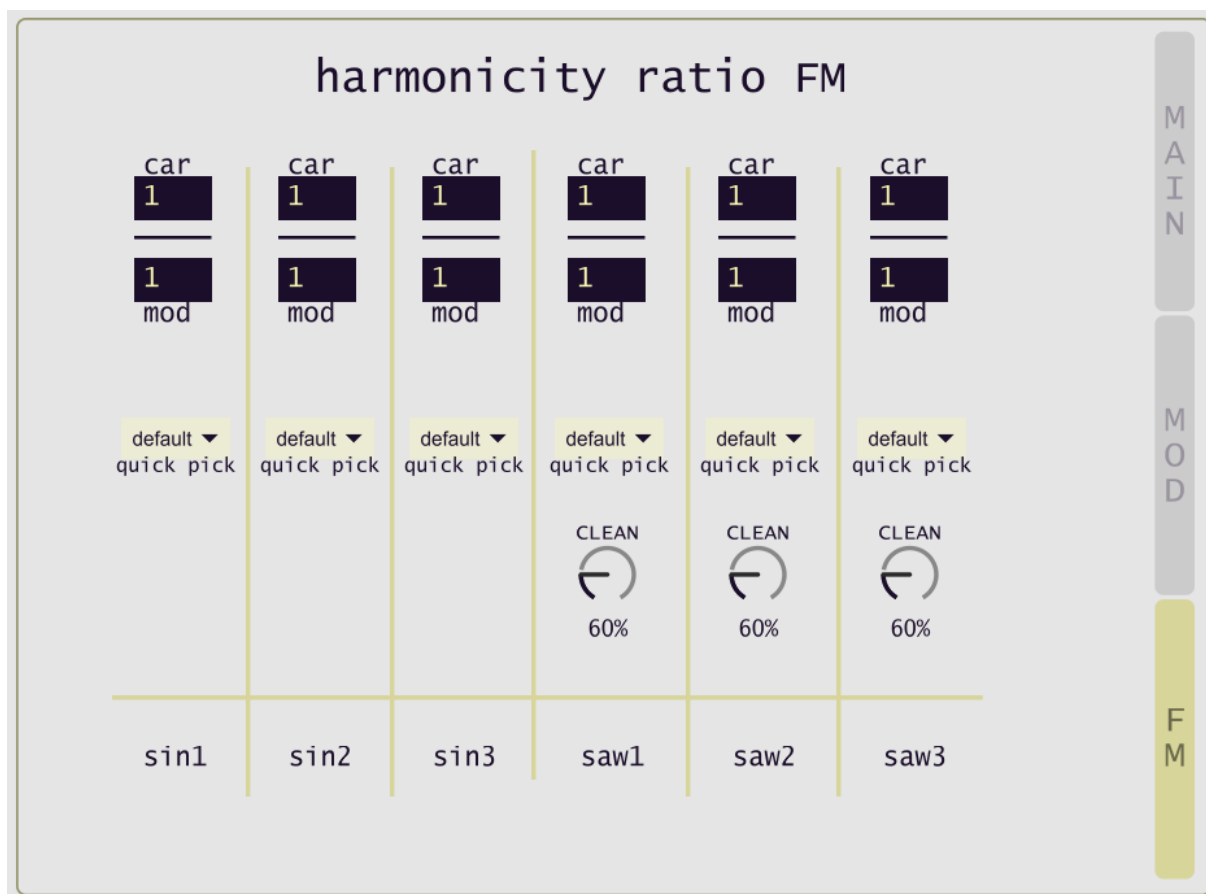


Figura 24

La terza e ultima schermata *FM* è un ulteriore grado di personalizzazione per la creazione del timbro per la sintesi in modulazione di frequenza e riferisce perciò ai primi due gruppi escludendo le onde quadre. Tramite questo pannello è possibile impostare l'*harmonicity ratio* citato prima, in modo da esprimerlo come rapporto fra la portante e la modulante. La frequenza della portante è determinata tramite i processi di assegnamento del sequencer, mentre quella della modulante è dedotta proprio da questo rapporto. Il menu *quick pick* offre una lista di rapporti intervallari della scala naturale, per impostare rapidamente una proporzione. Nel caso delle onde a dente di sega si può impostare l'efficacia della rimozione o attenuazione delle componenti in bassa frequenza tramite *CLEAN*, espresso in valori percentuali, dove valori più alti accentuano l'effetto simil passa alto.

## 2.3 Master section

In questa sezione sono presenti dei semplici controlli generali.



Figura 25

Qui è presente:

- un tremolo sull'uscita dello strumento di cui è specificabile l'efficacia
- il controllo del glide e della trasposizione in cents per ogni oscillatore
- un pannello per la registrazione dell'uscita audio

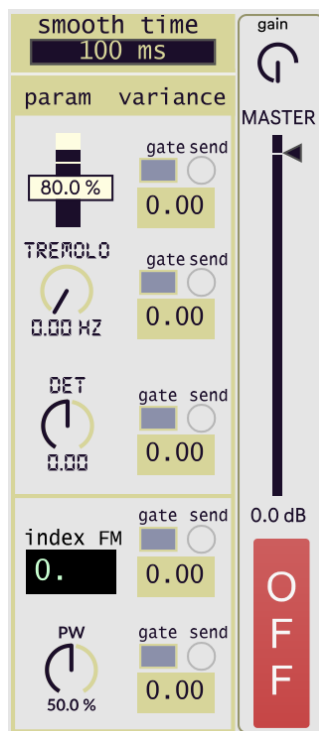


Figura 26

Parametri collettivi e controlli di accensione

Come si può notare in Figura 25 si ha un tremolo sull'uscita del *Coreeno* che dispone anche di un comando per l'efficacia di modulazione (*depth*). Oltre al traspositore in cents, c'è un controllo per il GLIDE, espresso in percentuale rispetto al valore ritmico in corso, che produce dei glissandi fra una frequenza e la successiva, per ogni oscillatore. In Figura 26: sulla destra è presente il volume master, con un eventuale gain di sostegno, e il pulsante ON/OFF del volume. Sulla sinistra ci sono dei parametri collettivi, che sono graficamente gli stessi della sezione *MAIN* della *synthesis section* e sono preposti proprio al loro controllo.

Quando il pulsante *gate* è attivo, il valore del parametro collettivo impostato viene inviato ai relativi parametri (quando non è attivo, non viene trasmesso nessun valore). Il valore

effettivamente ricevuto dagli oscillatori è determinato dalla varianza (*variance*) indicata accanto: se  $A$  è il valore di un certo parametro collettivo e  $B$  la sua varianza, i relativi parametri degli oscillatori possono assumere un valore compreso fra  $(A - B)$  e  $(A + B)$ . Il tempo di transizione da un valore a un altro è pari allo *smooth time* stabilito. Se il pulsante *gate* è spento è comunque possibile inviare istantaneamente il valore del parametro tramite il pulsante *set*.

## 2.4 Possibili estensioni e miglioramenti

Nel corso della trattazione sono state citate alcune delle esperienze significative sul tema della composizione automatizzata e della composizione assistita, ponendo l'accento sui modelli e i procedimenti più efficaci oltre agli strumenti matematici più adeguati in tal senso. Alcuni dei concetti sono stati inseriti all'interno di *Coreeno*, con l'obiettivo di realizzare un ambiente che possa integrare questo approccio con la sintesi sonora. Sono state fatte delle scelte riguardo a cosa rendere personalizzabile cercando di mantenere sempre un discreto controllo in tempo reale e offrire un utilizzo semplificato per quanto riguarda alcuni meccanismi. La filosofia alla base di questo sistema è di offrire una quantità limitata di informazioni, rinunciando a un controllo completo in favore di un'organizzazione generale.

*Coreeno* è programmabile in modo da produrre della musica in modo automatico ma richiede interventi costanti nell'intenzione di dar valore al risultato e variare gli elementi in gioco. Le tabelle coprono cicli di settantacinque eventi e quindi una modifica al momento opportuno può portare a cambiamenti evidenti ma abbastanza equilibrati. Questo *tool* è come un'equazione il cui risultato può subire cambiamenti sostanziali a fronte di piccole variazioni dei valori inseriti (cioè dei parametri in generale).

Per esempio, nella parte principale della *Synthesis section* è controllabile il solo indice di modulazione per la FM mentre l'*harmonicity ratio* ha un pannello apposito: questo è esplicativo dell'approccio che si è seguito in fase di progettazione, cioè meno parametri di controllo ma più efficaci. Nel caso di comandi importanti ma molto specifici si è preferito inserirli in apposite sottosezioni, non inizialmente esposte, in modo da favorire un approccio a delle macro-variazioni tramite i parametri più sostanziali. Parametri secondari e di maggiore specificità sono stati invece inseriti in interfacce dedicate, non immediatamente raggiungibili e volte quindi a un uso più approfondito. Le modulazioni sono un altro esempio di questo approccio poiché non permettono la realizzazione di ogni possibile comportamento ma la loro semplicità garantisce un utilizzo chiaro ed efficace.

Questi esempi di scelte sono di carattere progettuale, arbitrario, dettate da vincoli di praticità di utilizzo, ma comunque bisogna tener conto di altre restrizioni, come l'effettiva utilizzabilità in termini di risorse hardware. Si sono resi necessari alcuni compromessi in termini di utilizzo della CPU che in un ambiente come Max si riducono a limitare le due categorie audio e video. Ad esempio, il numero massimo di voci disponibili in contemporanea è limitato a venti (cioè venti sono le istanze disponibili per il poly~ nella realizzazione su Max) e gli oggetti grafici sono stati evitati il più possibile (spettrogrammi, spettroscopi, ecc.).

Un successivo e ulteriore lavoro è possibile in merito all'interfaccia, l'interattività e l'implementazione di nuove funzionalità. A compendio della fase di sviluppo è prevista una fase sistematica di beta testing con il maggior numero possibile di utilizzatori, che attraverso i loro riscontri possano ispirare nuove aggiunte o correzioni. Relativamente all'interfaccia, è possibile un miglioramento dell'aspetto grafico, cioè riguardo il posizionamento degli oggetti, la loro presentazione e le scelte stilistiche.

Il tema del controllo e dell'interattività è essenziale in uno strumento: periferiche midi, scorciatoie da tastiera e funzionalità rapide possono aiutare a migliorare l'usabilità.

Ci sono delle funzionalità che potrebbero essere implementate in versioni future, attualmente in fase di studio ed esplorazione. Tra le possibili espansioni che meritano menzione in quanto di sicura utilità per fini compositivi si elencano: funzionalità di registrazione, ripetizione e salvataggio di sequenze prodotte, sia in tempo reale ma anche per salvare su file delle sequenze midi. Si potrebbe aggiungere la memorizzazione su disco delle sequenze per riutilizzarle con o senza gli stessi rapporti ritmici, così che ogni volta verrebbero generate delle sequenze simili ma diverse, avendo una stessa successione di frequenze ma involuppi, volumi e parametri diversi.

Un'altra funzionalità aggiuntiva da menzionare sarebbe l'inserimento di altri modi di gestione dell'insieme delle frequenze, oltre a *scroll* e *pick*.

Nell'utilizzo automatico, può essere interessante offrire la possibilità di scrittura di sequenze di parametri, come delle macro-sequenze.

Nonostante ci sia margine per delle estensioni, lo strumento raggiunge gli obiettivi posti in fase di ideazione, riuscendo a mantenere un equilibrio fra semplicità e profondità, e rispettando i principi alla base del progetto.

## Capitolo 3 – produzione artistica (risultati sperimentali)

In questo capitolo vengono discussi alcuni esempi d'uso del *Coreeno*, in riferimento a contenuti multimediali forniti assieme a questa trattazione e reperibili all'indirizzo <https://github.com/Borjalito/Coreeno-media>. Dato l'intento espositivo di questo capitolo sono stati scelti degli elementi peculiari dello strumento e sono quindi proposte delle sequenze sonore adatte alla presentazione dello strumento.

*Coreeno* è organizzato per sezioni (sequencer, synthesis, master) in modo da suggerirne un uso per compartimenti. All'interno di queste sezioni sono presenti delle schermate e alcune fra queste presentano diverse pagine: all'avvio lo strumento è impostato in modo da offrire le schermate fondamentali ed evitare la gestione di troppe informazioni, inizialmente superflue, che potrebbero disorientare.

### Video esplicativi

Viene allegata una serie di tre video esplicativi con l'intento di mostrare un esempio di utilizzo tramite le schermate presenti all'avvio. Ogni video mostra un esempio d'uso di una sezione in particolare da cui si può notare l'effetto di alcuni parametri.

### Esempi audio

Per le registrazioni degli esempi sono stati imposti dei modi di gestione dei parametri dello strumento. Vengono elencate possibili istruzioni sull'uso dei parametri imposte durante la realizzazione degli esempi proposti:

-Sequencer: tonale/microtonale, millisecondi/BPM, scala/e utilizzata/e, velocità metronomo, figure/valori ritmici, probabilità (PROB), registro (FUND, OCT), salti intervallari (VAR)

-Modalità: single/multi con parametri

-Inviluppo: durata totale, durata segmenti

-Tabelle: sustain e velocity

-Gruppi oscillatori (sin, saw, rect): volume, filtro, modulazioni

-Quantità, modalità d'uso e tipologia parametri oscillatori

-Quantità e modalità d'uso modulazioni per parametri oscillatori

-Impostazioni FM harmonicity ratio

Questo è di un elenco delle eventuali configurazioni per lo strumento e sono possibili imposizioni più o meno stringenti su questi parametri al fine di sperimentare con una certa cognizione. Per gli esempi, quindi, le istruzioni sono di volta in volta selezionate tramite certi criteri di ricerca e possono fornire indicazioni più o meno specifiche. Di seguito le tabelle riassuntive e i commenti relativi agli esempi proposti.

SEQUENCER	Sequencer BPM 80 Scala SI minore armonica con armonizzazione a 3 o 4 voci su scala Inviluppo variabile (lungo per accordi e corto per melodie)
OSCILLATORI	Per accordi solo sinusoidi e dente di sega Per il resto tutti i gruppi Uso di CLEAN e specificazione
MODULAZIONI	Nulle per gli accordi Leggere per il resto su FM e PW

*Tabella Esempio 1*

Esempio 1 (candidato): per questa realizzazione si sono selezionati quattro accordi dall'armonizzazione della scala di si minore: re aumentato, si minore con settima maggiore,

fa# con settima minore, la# diminuito. Questa sequenza è stata registrata con inviluppo disteso per ottenere la traccia d'accompagnamento e si è in seguito generata una melodia usando la scala di riferimento, stavolta con inviluppo corto per creare una melodia staccata. L'intento di questo esempio è di esporre le possibilità dello strumento e non di ottenere un risultato sonoro particolare, in quanto non sono state applicate modificazioni di alcun tipo.

SEQUENCER	Fondamentale RE Scala minore napoletana Metronomo 60 BPM PROB intorno al 50% per le sequenze ritmate e variabile altrimenti Figure ritmiche: semiminima, croma, semicroma Tabelle prevalentemente fisse (velocity 127 e sustain 40%) Inviluppo con attacco corto e segmenti restanti variabili
OSCILLATORI	Solo sinusoidi e quadre (focus su FM e PW)
MODULAZIONI	Presenti su FM e PW, in minima quantità sui tremoli

*Tabella Esempio 2*

Esempio 2 (candidato): per questa realizzazione sono stati fissati alcuni parametri arbitrari come punto di partenza della sperimentazione e si è lavorato sulle restanti opzioni, allo scopo di ottenere sequenze di diverso tipo in termini di occupazione dello spettro, tipologia d'inviluppo, rapidità d'esecuzione. In questo modo si è realizzata una composizione per sovrapposizione dei diversi materiali senza alcuna trasformazione, a eccezione di volumi e tagli per creare le diverse tracce audio.



SEQUENCER	Fondamentale variabile su frequenze medio basse Muti mode PARTS=24, SIZE=3 OCT e VAR variabile intorno a, rispettivamente, 5 e 2 Sequencer microtonale, BPM 70 PROB variabile con prevalenza bassa (intorno al 20-30 %) Figure ritmiche variabili fra croma, semicroma, semiminima e le relative in terzina Tabelle dinamiche Inviluppo con attacco e release lunghi, sustain variabile da tabelle e tempo di dominio variabile ma attorno a 1 secondo
OSCILLATORI	Tutti i gruppi utilizzati (sinusoidi, dente di sega, quadre) con modulazioni lievi sui rispettivi volumi Utilizzo accentuato tremolo Utilizzo lieve detune Utilizzo FM, in particolare sulle onde a dente di sega con CLEAN al 100%
MODULAZIONI	Accentuate sui tremoli e lievi sulla FM

*Tabella Esempio 3*

Esempio 3 (candidato): per questa realizzazione si è sperimentato utilizzando una scala microtonale con PARTS=24 e ridisegnata per avere alcuni gradi della scala vicini fra loro, così da creare battimenti. L'intento di questo studio è di creare timbri basandosi sul controllo dell'ampiezza degli oscillatori (tremolo) e mescolare gli effetti ottenuti con dei battimenti generati fra i gradi con frequenze vicine sulla scala, oltre che da lievi detune posti su alcuni oscillatori. Avendo infatti un parametro VAR molto basso, è probabile ottenere due estrazioni successive con la stessa frequenza e avere così due oscillatori contigui con stessa frequenza.

SEQUENCER	Multi mode OCT=3 e VAR intorno ai 3 Sequencer tonale, scala YO, BPM 120, PROB 100% Figure ritmiche senza terzina Tabelle prevalentemente fisse Inviluppo con attacco corto
OSCILLATORI	Tutti i gruppi utilizzati (sinusoidi, dente di sega, quadre) per suddivisione dello spettro. Uso del tremolo sia negli oscillatori che sul master
MODULAZIONI	Lievi su FM

*Tabella Esempio 4*

Esempio 4 (candidato): per questa realizzazione si è lavorato sulla scala YO, appartenente alla tradizione giapponese, con la ricerca di un suono pulito e rilassante. L'esempio è stato ottenuto per sovrapposizioni di diverse tracce.

Sono stati inoltre preparati degli esempi sonori grazie al contributo di una persona esterna, Gioele Billi, che ha offerto la propria disponibilità per l'esecuzione e la registrazione di varie sequenze, da cui sono stati tratti dei frammenti curati dal candidato.

SEQUENCER	Fondamentale variabile (sequenza suonata da tastiera) Multi mode OCT=1, VAR=1-2 Sequencer tonale, BPM 180 PROB intorno al 100% ma variabile Figure ritmiche variabili, non in terzina Tabelle fisse Inviluppo con attacco corto e release per risonanze
OSCILLATORI	Tutti i gruppi utilizzati (sinusoidi, dente di sega, quadre) con modulazioni evidenti sui rispettivi volumi Ampio utilizzo parametri oscillatori
MODULAZIONI	Molto presenti mediante onda a dente di sega con frequenza sostenuta (

*Tabella Esempio 1 Gioele*

Esempio 1 (Gioele): per questa realizzazione è stato lasciato ampio grado di libertà e, soprattutto tramite le modulazioni, si è ricaduti in un caso particolare di risonanze ed effetti percussivi. Non sono state usate tracce diverse e quindi sovrapposizioni di alcun tipo.

SEQUENCER	BPM 125 Multi mode OCT=2, VAR=1 Nessuna scala (una nota su più ottave) Inviluppo con attacco e release corti e spostamenti sul decay Semiminima e PROB al 100% Tabella della velocity variabile con circa metà degli eventi a 127 e i restanti su valori più bassi
OSCILLATORI	Tutti attivi con attenzione alla divisione dello spettro fra i diversi gruppi, utilizzando CLEAN per le FM delle onde a dente di sega.
MODULAZIONI	Utilizzo lieve delle modulazioni

*Tabella Esempio 2 Gioele*

Esempio 2 (Gioele): per questa realizzazione è stata usata una fondamentale fissa riproposta su ottave diverse, con attacco corto per generare una ripetizione percussiva ostinata. Sono state sovrapposte varie tracce per ottenere un crescendo e un incastro ritmico sul finale, con accumulazione degli elementi.

## Conclusione

Futuri rilasci di nuove versioni potranno approfondire aspetti del *Coreeno* e offrire nuove funzionalità. I risultati ottenuti sono in linea con le aspettative nella realizzazione di uno strumento capace di evolvere i principi alla base della progettazione e nel soddisfacimento dell'aspetto sperimentale. Un utilizzo da parte di molti utenti e il raccoglimento del loro riscontro può portare miglioramenti negli aspetti meno indagati per porre l'accento su ciò che esprime al meglio le possibilità di *Coreeno*.

# Riferimenti

**Ader Lidia, Stefanija Leon e Stanevičiūtė Rūta** Microtonal Music in Central and Eastern europe: Historical Outlines and Current Practices [Rapporto]. - Ljubljana : Univerza v Ljubljani, Filozofska fakulteta, 2020.

**Ames Charles** Automated Composition in Retrospect [Journal] // Leonardo, Vol. 20, No. 2, Special Issue: Visual Art, Sound, Music and Technology. - 1987. - p. 169-185.

**Bartók Béla** Scritti sulla musica popolare [Libro]. - [s.l.] : Bollati Boringhieri, 1997.

**botnikstudios** New Chapter of Harry Potter [Online]. - 2018. - <https://botnik.org/content/harry-potter.html>.

**Chierotti Carlo Mario** <https://kircher.chierotti.it/> [Online]. - <https://kircher.chierotti.it/>.

**Cycling '74, Documentazione Max** Max 8 Documentation [Online]. - <https://docs.cycling74.com/max8>.

**Di Nunzio Alex** Groove [Online]. - 02 02 2011. - <https://www.musicainformatica.it/argomenti/groove.php>.

**Google, Imagen** imagen [Online]. - <https://imagen.research.google/>.

**IBM, Reti Neurali** Reti neurali [Online]. - <https://www.ibm.com/it-it/cloud/learn/neural-networks>.

**ictbusiness** Creare video da un testo. Ora è possibile con Meta. [Online]. - 30 Settembre 2022. - <https://www.ictbusiness.it/cont/news/creare-video-da-un-testo-ora-e-possibile-con-meta/47064/1.html#.YzwVY0xBy5c>.

**ilpost** C'è un software di intelligenza artificiale che inventa storie piuttosto pazzesche [Online]. - 09 Febbraio 2022. - <https://www.ilpost.it/flashes/software-intelligenza-artificiale-storie/>.

**Manfrin Luigi** L'immagine spettrale del suono e l'incarnazione del tempo allo stato puro: la teoria della forma musicale negli scritti di Gérard Grisey [Journal]. - [s.l.] : Libreria Musicale Italiana (LIM), 2003. - 1 : Vol. 38.

**Maurier John A.** A Brief History of Algorithmic Composition [Online]. - Marzo 1999. - <https://ccrma.stanford.edu/~blackrse/algorithm.html>.

**OpenAI DALL-E-2** DALL-E-2 [Online]. - <https://openai.com/dall-e-2/>.

**Oracle, IA** Intelligenza artificiale [Online]. - <https://www.oracle.com/it/artificial-intelligence/what-is-ai/>.

**Oracle, Machine Learning** Machine Learning [Online]. - <https://www.oracle.com/it/artificial-intelligence/machine-learning/what-is-machine-learning/>.

**Palisca Claude V. e Pesce Dolores** Guido of Arezzo, Grove Music Online [Online]. - 20 Gennaio 2001. - <https://doi.org/10.1093/gmo/9781561592630.article.11968>.

**Simonetta Federico** Modellizzazione musicale tramite catene di Markov [Rapporto]. - Pavia : Tesi facoltà di ingegneria elettrica e informatica, 2014.

**Treccani, Processo Stocastico** Definizione "processo stocastico" [Online]. - [https://www.treccani.it/enciclopedia/processo-aleatorio\\_%28Dizionario-di-Economia-e-Finanza%29/](https://www.treccani.it/enciclopedia/processo-aleatorio_%28Dizionario-di-Economia-e-Finanza%29/).

## Ringraziamenti

Ringrazio il professor Canavese e la professoressa Federici per l'aiuto e per i suggerimenti forniti, per la competenza dimostrata e il tempo dedicato.

Ringrazio Anastasiya e la mia famiglia per il supporto fornito e per il ruolo che hanno avuto nel completamento di questo percorso.

Ringrazio Gioele per gli spunti forniti in merito allo strumento, per le prove e le discussioni in merito, fin dalla fase embrionale del progetto.